Vorlesung Fehlerrechnung & Datenanalyse

Olaf Kaczmarek & Udo Werner

Heutiges Thema:

Daten, χ^2 –Fits, verschiedene Skalen

- Methode der kleinsten Quadrate → Ausgleichsgraden, χ^2-Fits
- Linearisierung von Daten
- Transformation von Daten und Fehlerfortpflanzung
- Logarithmische Skalen

Methode der kleinsten Quadrate

Gegeben seien:

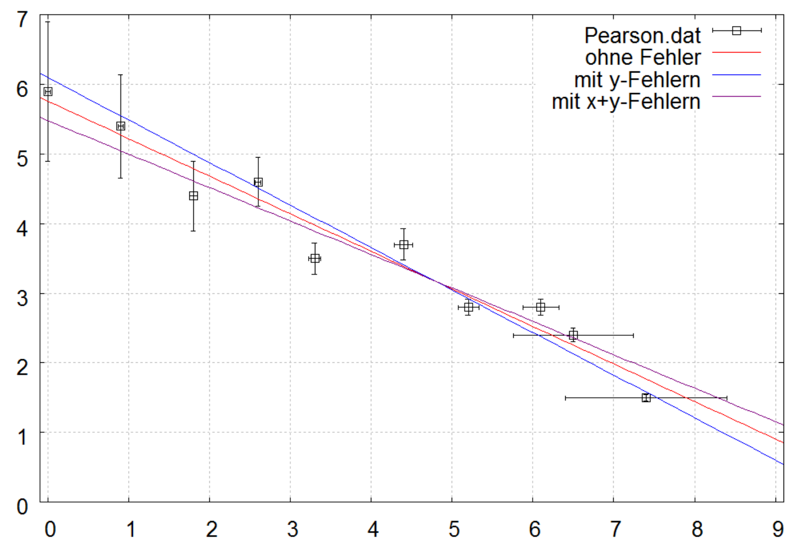
- Ein Satz von genau bekannten x-Werten $\{x_i\}$
- Ein Satz von gemessenen y-Werten $\{y_i\}$ mit Unsicherheiten $\{\sigma_i\}$
- Eine von Parametern $a=(a_1,\ldots,a_k)$ abhängige Funktion f(x;a), die die y-Werte vorhersagt

Ziel:

- **Die Parameter a_k so zu bestimmen, dass die Funktion f(x;a) den Datensatz möglichst gut beschreibt!**

Gütekriterium:

$$\chi^2 = \sum_i \left(\frac{y_i - f(x_i;a)}{\sigma_i} \right)^2$$



Methode der kleinsten Quadrate

Gütekriterium:

$$\chi^2 = \sum_i \left(\frac{y_i - f(x_i; a)}{\sigma_i} \right)^2$$

minimiere χ^2 → Gute Approximation ("Fit")

notwendige Bedingung:

$$\frac{\partial \chi^2}{\partial a_j} = 0 \qquad j = 1, .., k$$

→ Gleichungssystem für Bestimmung der a_j:

$$\sum_i \frac{1}{\sigma_i^2} \frac{\partial f(x_i; a)}{\partial a_j} (y_i - f(x_i; a)) = 0 \qquad j = 1, ..k$$

Least squares fit: $y = m\,x + b$

Messpunkte $(x_1, y_1 \pm \sigma_y)$, ... $(x_N, y_N \pm \sigma_y)$ d.h. zunächst konstante Fehler $\sigma_i = \sigma_y$

Minimiere $$\chi^2 = \sum_{i=1}^{N} \frac{(y_i - m x_i - b)^2}{\sigma_y^2}$$

Notwendige Bedingung

$$0 = \frac{\partial \chi^2}{\partial b} = \left(\frac{-2}{\sigma_y^2}\right) \sum_{i=1}^{N} (y_i - m x_i - b)$$

$$0 = \frac{\partial \chi^2}{\partial m} = \left(\frac{-2}{\sigma_y^2}\right) \sum_{i=1}^{N} x_i (y_i - m x_i - b)$$

$$\Longrightarrow \begin{aligned} \sum y_i &= N b + m \sum x_i \\ \sum x_i y_i &= b \sum x_i + m \sum x_i^2 \end{aligned} \Longrightarrow \begin{aligned} b &= \frac{(\sum x_i^2)(\sum y_i) - (\sum x_i)(\sum x_i y_i)}{N \sum x_i^2 - (\sum x_i)^2} \\ m &= \frac{N(\sum x_i y_i) - (\sum x_i)(\sum y_i)}{N \sum x_i^2 - (\sum x_i)^2} \end{aligned}$$

Least squares fit: y = m x + b

Messpunkte $(x_1, y_1 \pm \sigma_y), \ldots (x_N, y_N \pm \sigma_y)$ d.h. zunächst konstante Fehler $\sigma_i = \sigma_y$

Minimiere
$$\chi^2 = \sum_{i=1}^{N} \frac{(y_i - m x_i - b)^2}{\sigma_y^2}$$

Lösung:
$$\hat{m} = \frac{\overline{xy} - \overline{x} \cdot \overline{y}}{\overline{x^2} - \overline{x}^2}$$
$$\hat{b} = \overline{y} - \hat{m}\overline{x}$$

Fehler:
$$\sigma_m^2 = \frac{\sigma^2}{N(\overline{x^2} - \overline{x}^2)}$$
$$\sigma_b^2 = \frac{\sigma^2 \overline{x^2}}{N(\overline{x^2} - \overline{x}^2)}$$

Least squares fit: y = m x + b

Messpunkte $(x_1, y_1 \pm \sigma_1), \ldots (x_N, y_N \pm \sigma_N)$

Minimiere
$$\chi^2 = \sum_{i=1}^{N} \frac{(y_i - m x_i - b)^2}{\sigma_i^2}$$

bei Verwendung der gewichteten Mittelwerte:

$$\overline{y} = \frac{\sum y_i/\sigma_i^2}{\sum 1/\sigma_i^2} \qquad \text{und} \qquad \sigma^2 := \frac{\sum \sigma_i^2/\sigma_i^2}{\sum 1/\sigma_i^2} = \frac{N}{\sum 1/\sigma_i^2}$$

gilt wieder:

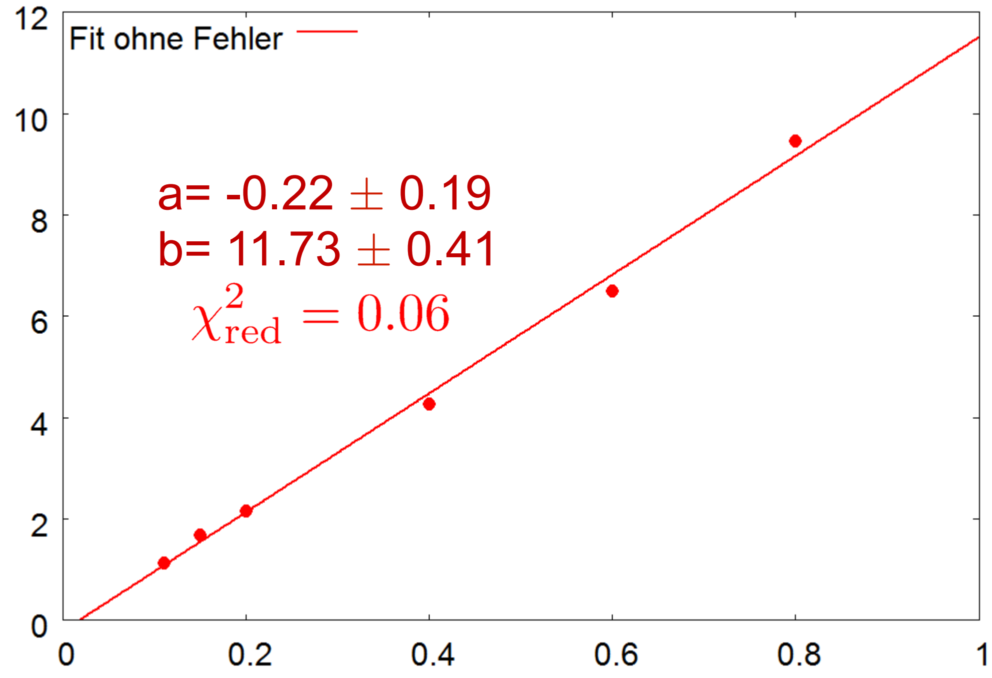
$$\hat{m} = \frac{\overline{xy} - \overline{x}\cdot\overline{y}}{\overline{x^2} - \overline{x}^2} \qquad \sigma_m^2 = \frac{\sigma^2}{N(\overline{x^2} - \overline{x}^2)}$$

$$\hat{b} = \overline{y} - \hat{m}\overline{x} \qquad \sigma_b^2 = \frac{\sigma^2\overline{x^2}}{N(\overline{x^2} - \overline{x}^2)}$$

Linearer Fit

Ohne Berücksichtigung der Unsicherheiten:

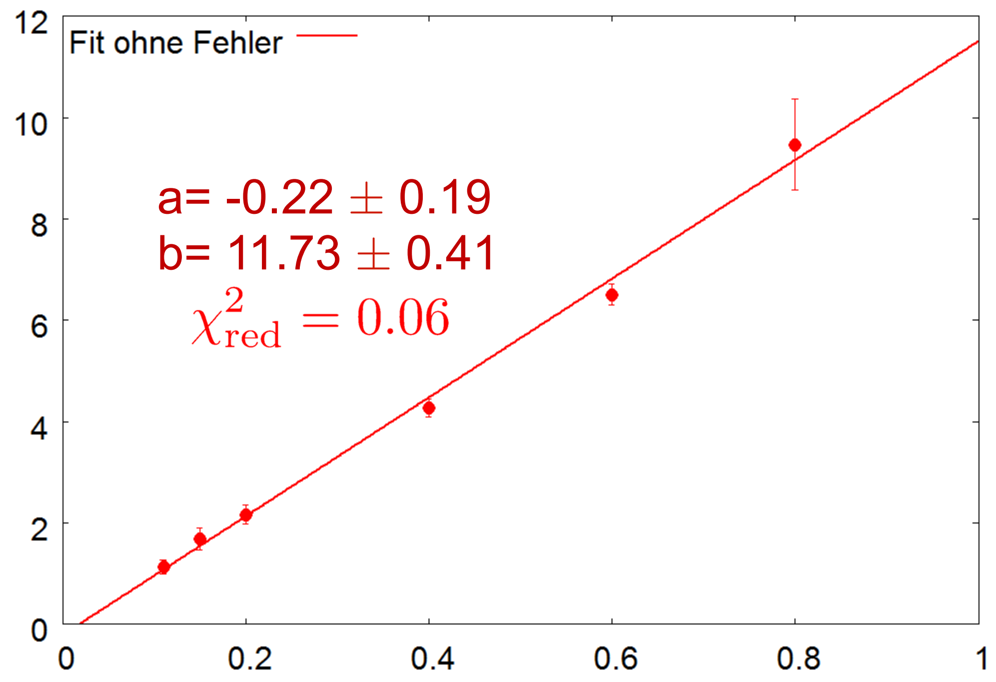
Fit mit f(x) = a + b*x



Linearer Fit

Ohne Berücksichtigung der Unsicherheiten:

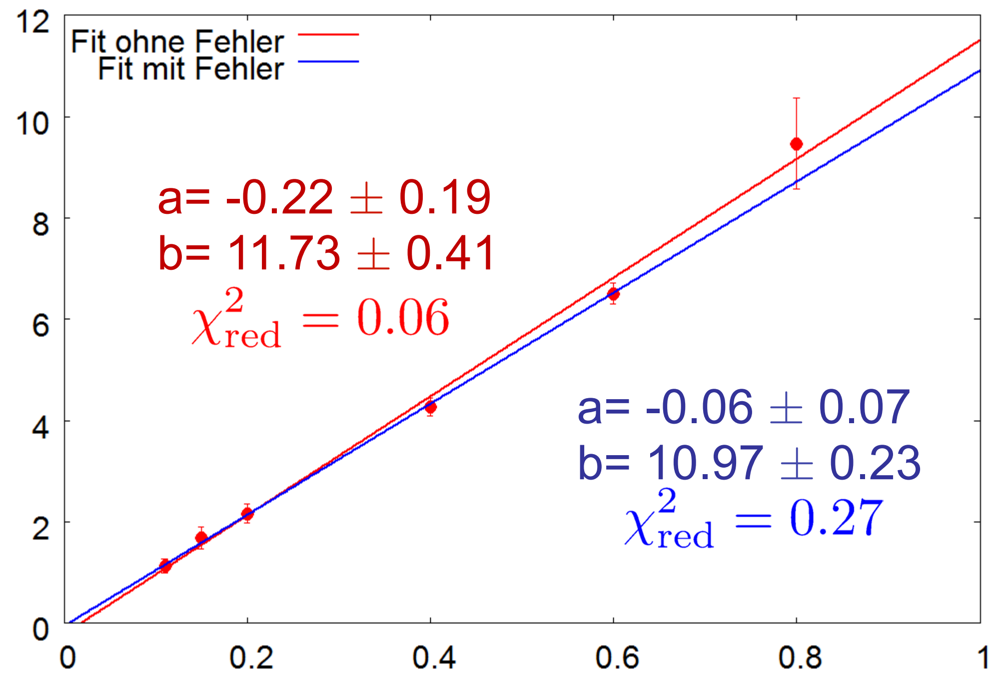
Fit mit f(x) = a + b*x



Linearer Fit

Mit Berücksichtigung der Unsicherheiten:

Fit mit f(x) = a + b*x



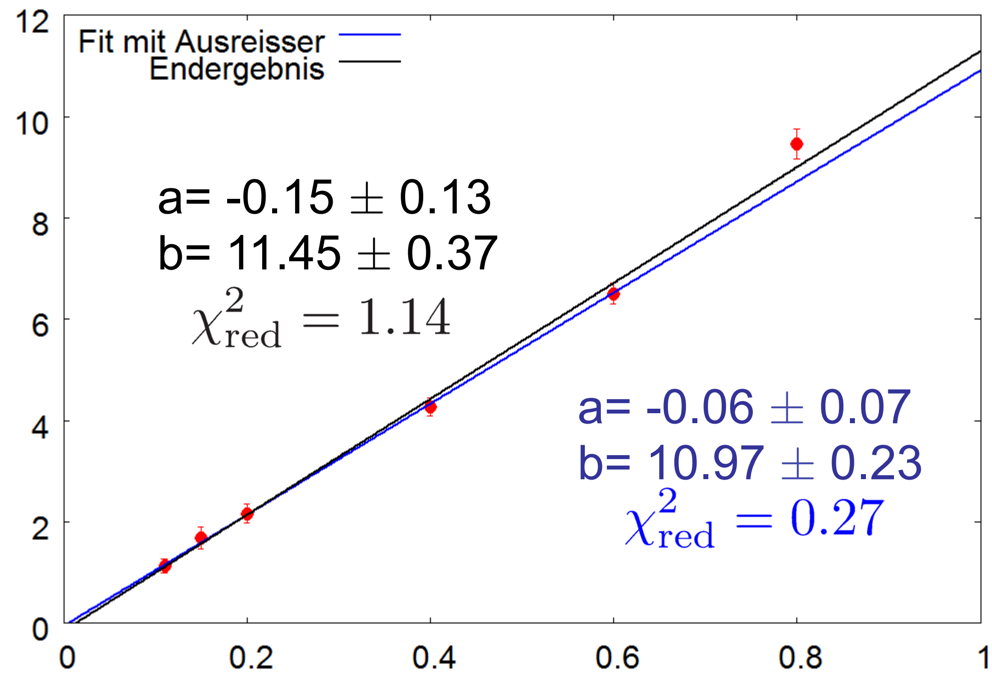
in gnuplot:

f(x) = a + b*x
fit f(x) “datenfile“ u 1:2 via a,b (ohne Fehler)
fit f(x) “datenfile“ u 1:2:3 via a,b (mit Fehler)

Linearer Fit

Mit Berücksichtigung der Unsicherheiten:

Fit mit f(x) = a + b*x



in gnuplot:

f(x) = a + b*x
fit f(x) “datenfile“ u 1:2 via a,b (ohne Fehler)
fit f(x) “datenfile“ u 1:2:3 via a,b (mit Fehler)

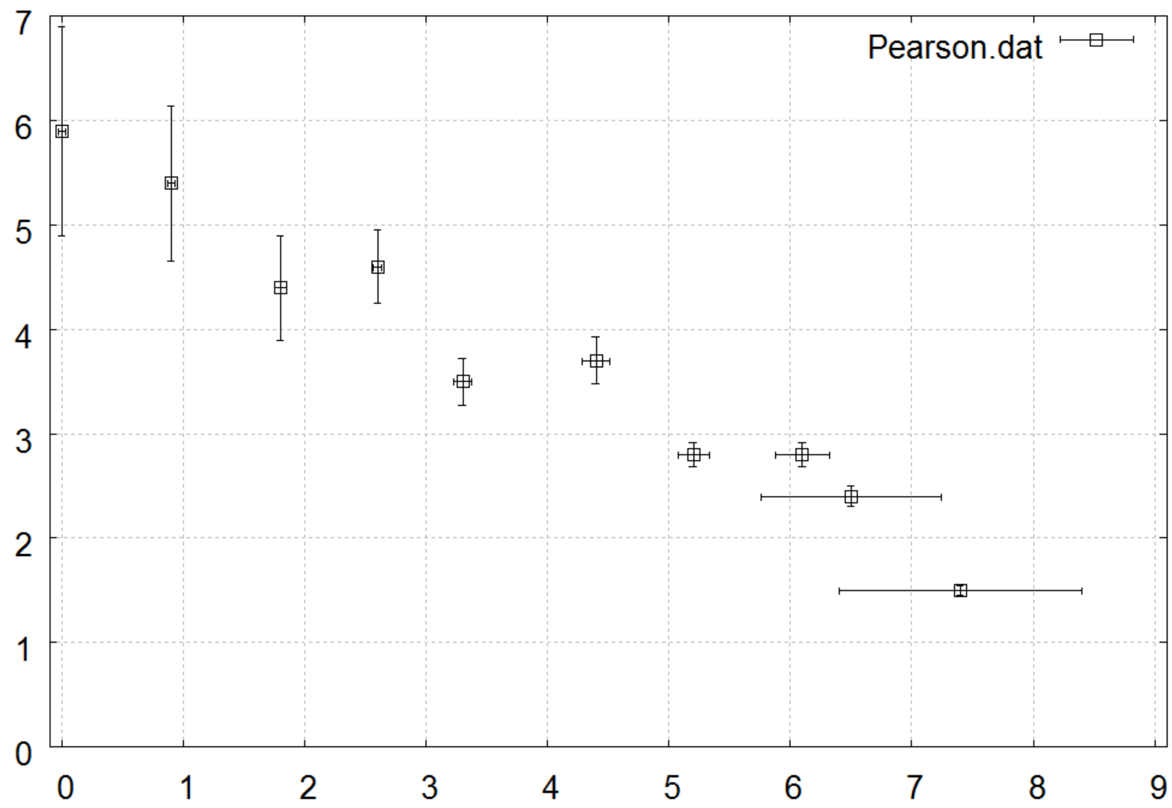
Beispiel Pearson.dat

Fit mit f(x) = m*x + b

FitLin(x,y):
ohne Fehler

FitLinDy(x,y,dy):
mit y-Fehlern

Gnuplot (mit y-Fehlern):



FitLinDxDy(x,y,dx,dy):
mit x+y-Fehlern

[matlab Routinen von Udo Werner siehe Homepage der Vorlesung]

Beispiel Pearson.dat

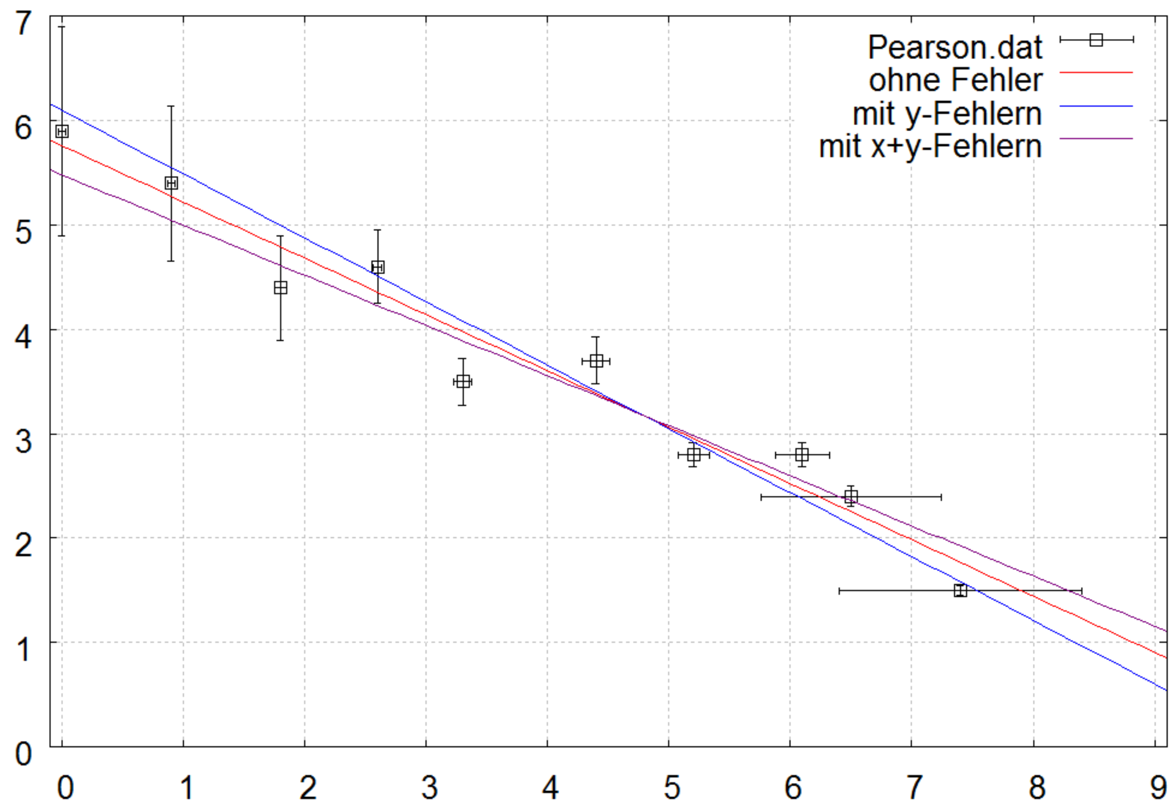
Fit mit f(x) = m*x + b

FitLin(x,y):
ohne Fehler
m1 = -0.54
b1 = 5.76

FitLinDy(x,y,dy):
mit y-Fehlern
m2 = -0.61 $\pm$ 0.03
b2 = 6.10 $\pm$ 0.20

Gnuplot (mit y-Fehlern):
m = -0.61 $\pm$ 0.06
b = 6.10 $\pm$ 0.42

FitLinDxDy(x,y,dx,dy):
mit x+y-Fehlern
m3 = -0.48 $\pm$ 0.06
b3 = 5.48 $\pm$ 0.29



Bester Fit unter Berücksichtigung aller Unsicherheiten:

f(x) = -0.48(6)*x + 5.48(29)

[matlab Routinen von Udo Werner siehe Homepage der Vorlesung]

Beispiel problem.dat

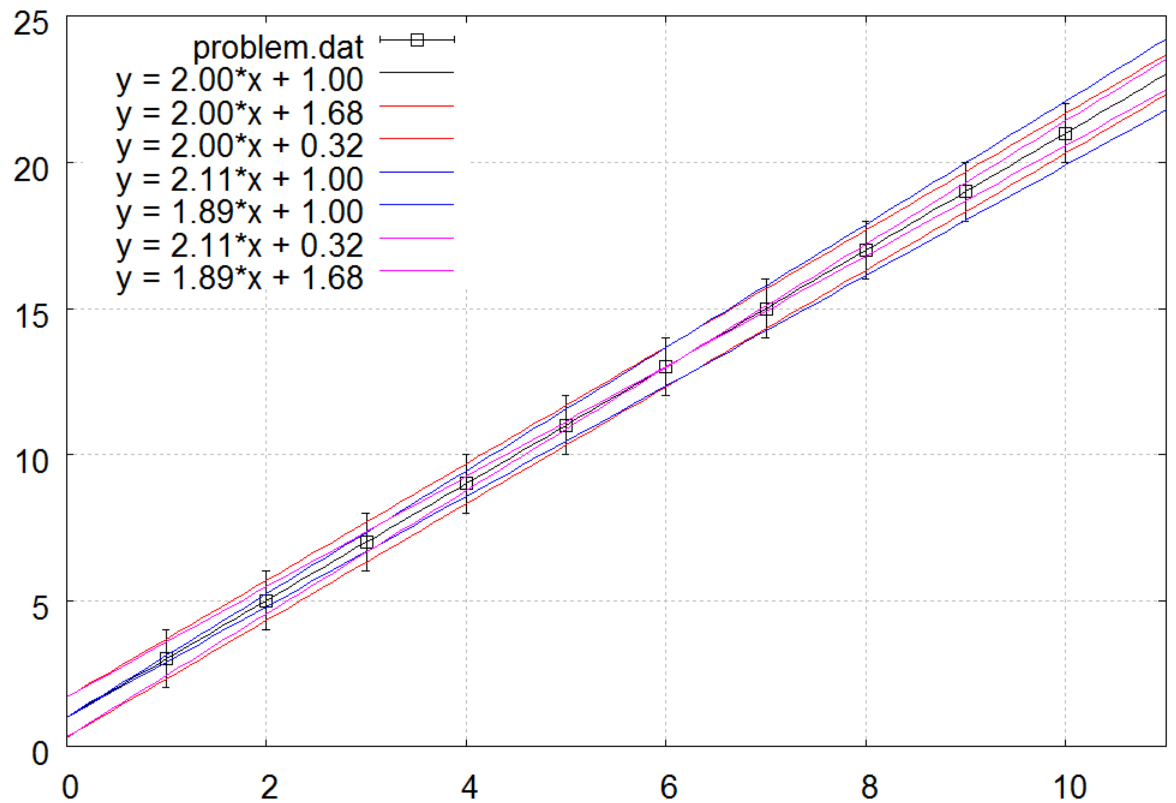
Fit mit f(x) = m*x + b

FitLinDy(x,y,dy):
mit y-Fehlern
$m = 2.00 \pm 0.11$
$b = 1.00 \pm 0.68$

Gnuplot:
mit y-Fehlern
$m = 2.00 \pm$???
$b = 1.00 \pm$???

Hmmmm....
Sum of squared residuals is zero. Can‘t compute errors

(andere Programme haben das gleiche Problem)



Analytisches Ergebnis:

$m = 2.00 \pm 0.11 \quad b=1.00 \pm 0.68$

Warum? $\chi^2 = 0$! gnuplot benutzt nicht die analytische Berechnung!

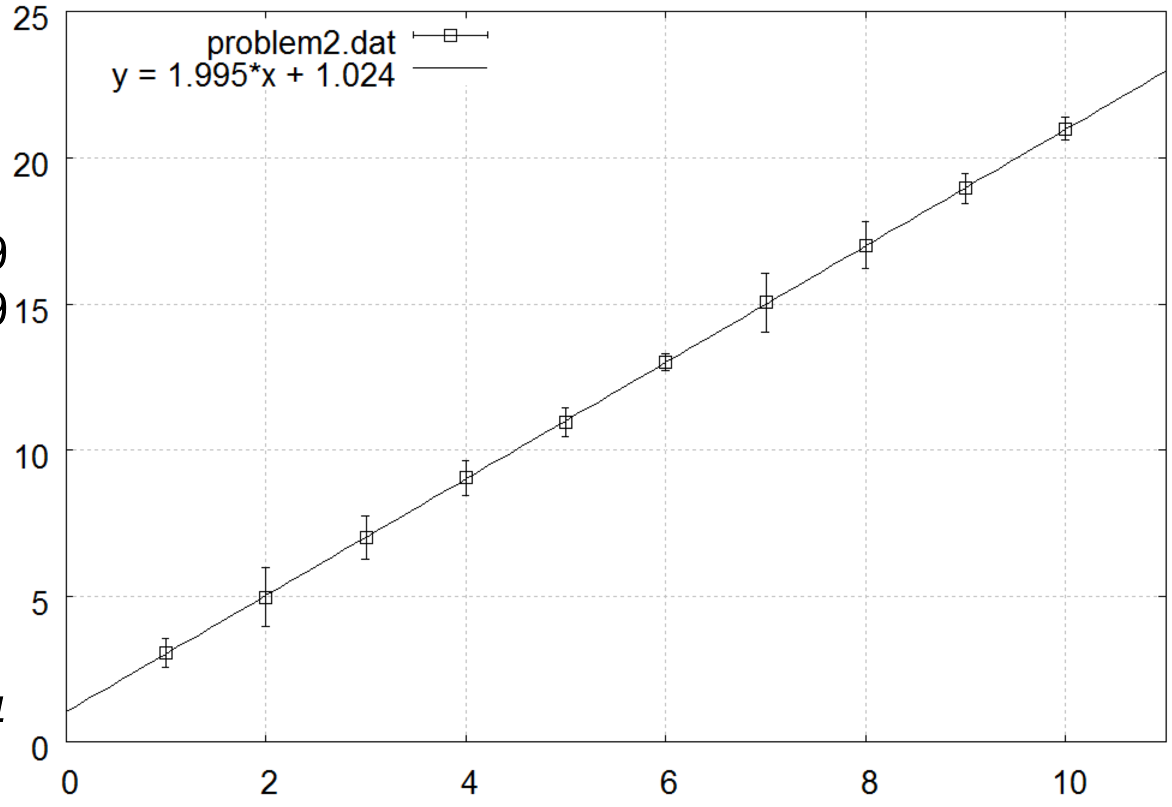
Beispiel problem2.dat

Fit mit f(x) = m*x + b

FitLinDy(x,y,dy):
mit y-Fehlern
m = 1.995093 ± 0.058699
b = 1.028104 ± 0.391409

Gnuplot:
mit y-Fehlern
m = 1.99509 ± 0.003875
b = 1.02810 ± 0.025840

reduced chisquare 0.0044



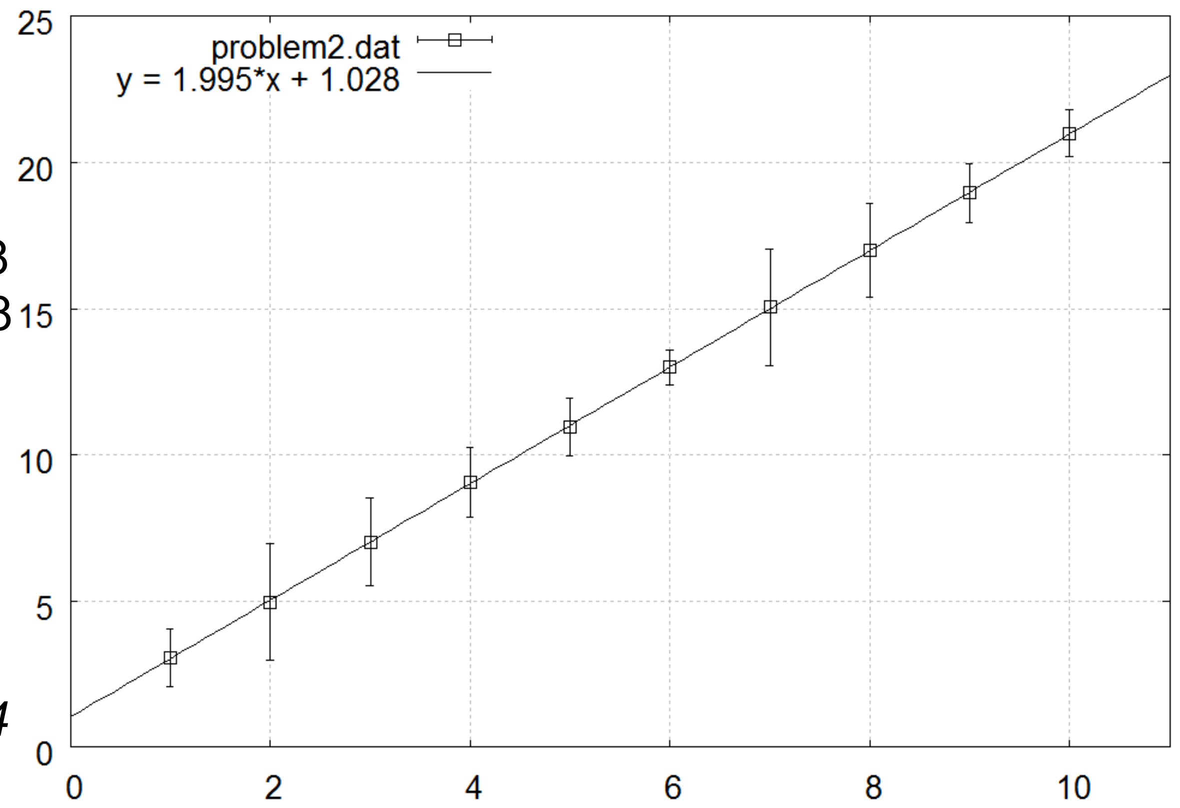
Beispiel problem2.dat

Fit mit f(x) = m*x + b

FitLinDy(x,y,dy):
mit y-Fehlern
m = 1.995093 ± 0.117398
b = 1.028104 ± 0.782818

Gnuplot:
mit y-Fehlern
m = 1.99509 ± 0.003875
b = 1.02810 ± 0.025840

reduced chisquare 0.0044

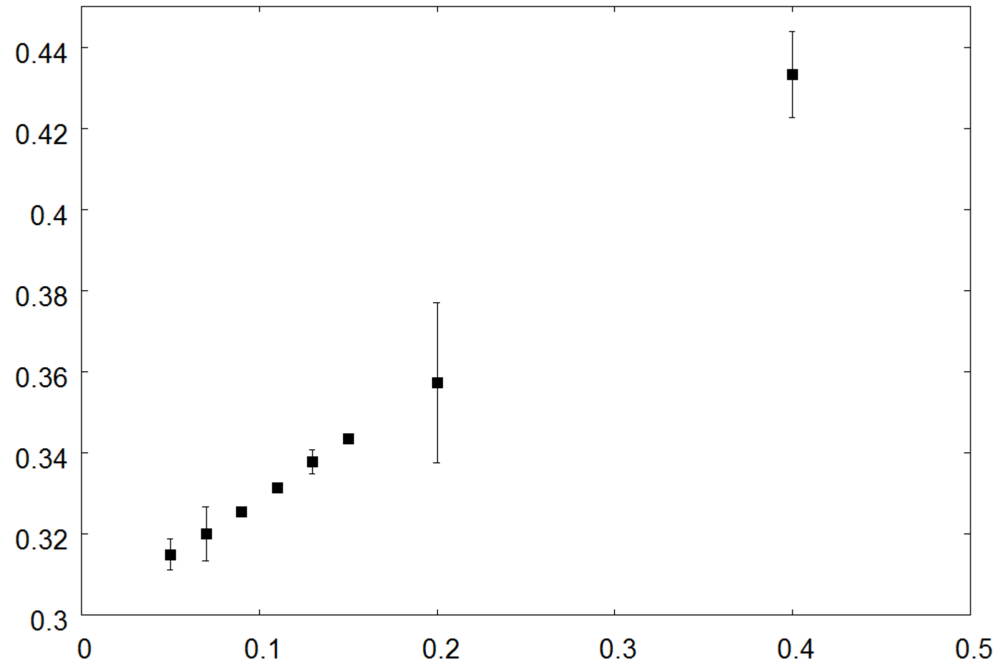


Verdoppelung der Fehler:
Fehler in der analytischen Rechnung doppelt so gross!
Fehler in gnuplot unverändert und unterschätzt!

Warum? χ^2 klein ! gnuplot benutzt nicht die analytische Berechnung!

Weiteres Beispiel

Zeigen diese Daten auch einen linearen Zusammenhang?

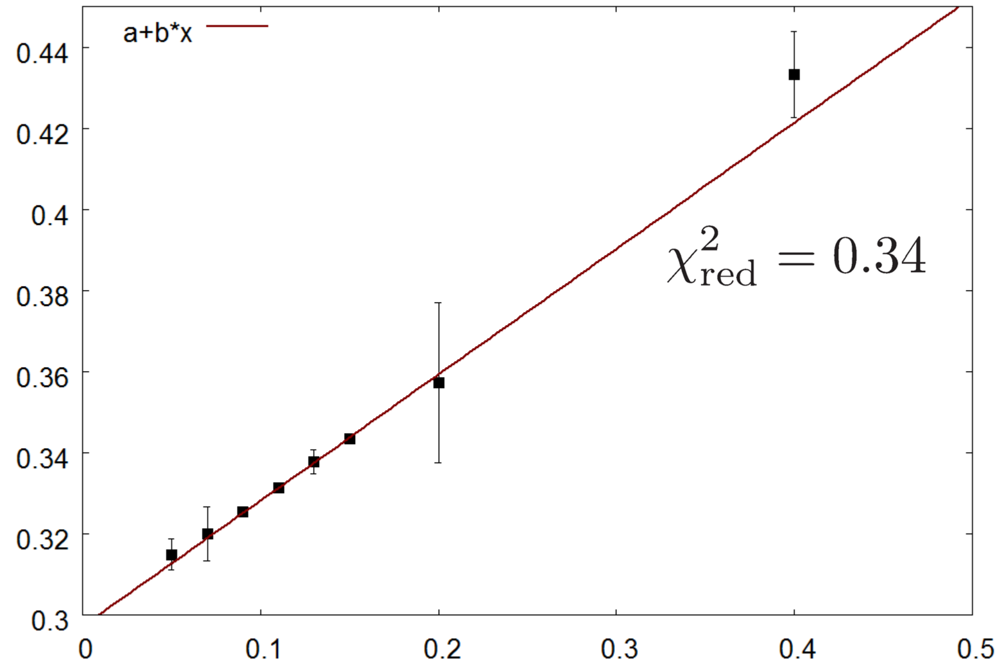


Weiteres Beispiel

Zeigen diese Daten auch einen linearen Zusammenhang?

f(x) = a + b*x

beschreibt die Daten recht gut!



Weiteres Beispiel

Zeigen diese Daten auch einen linearen Zusammenhang?

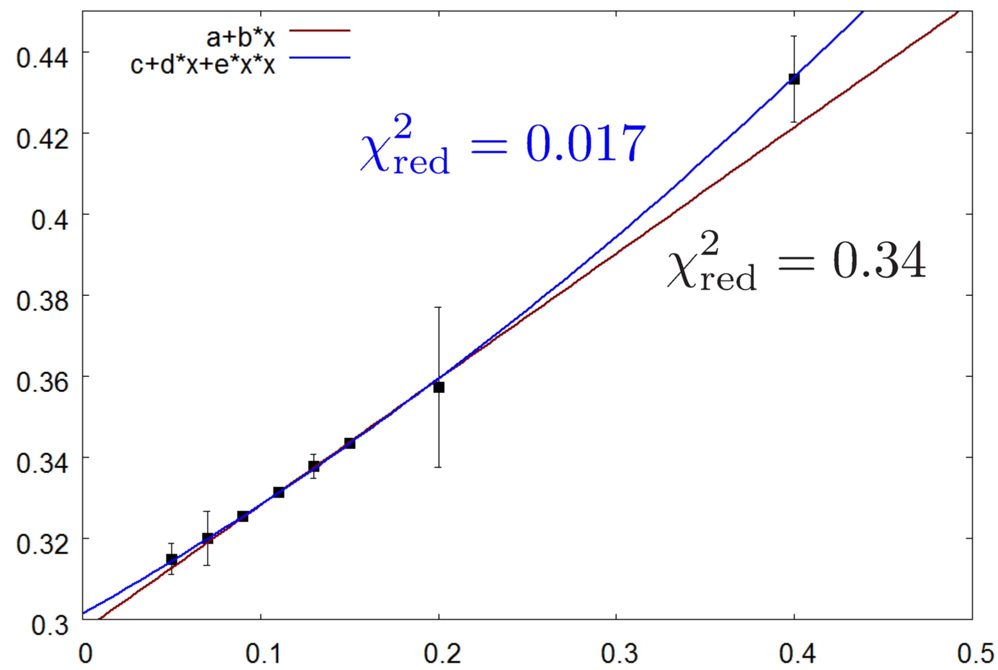
f(x) = a + b*x

beschreibt die Daten recht gut!

aber auch

g(x) = c + d*x + e*x*x

beschreibt die Daten recht gut!



Weiteres Beispiel

Zeigen diese Daten auch einen linearen Zusammenhang?

f(x) = a + b*x

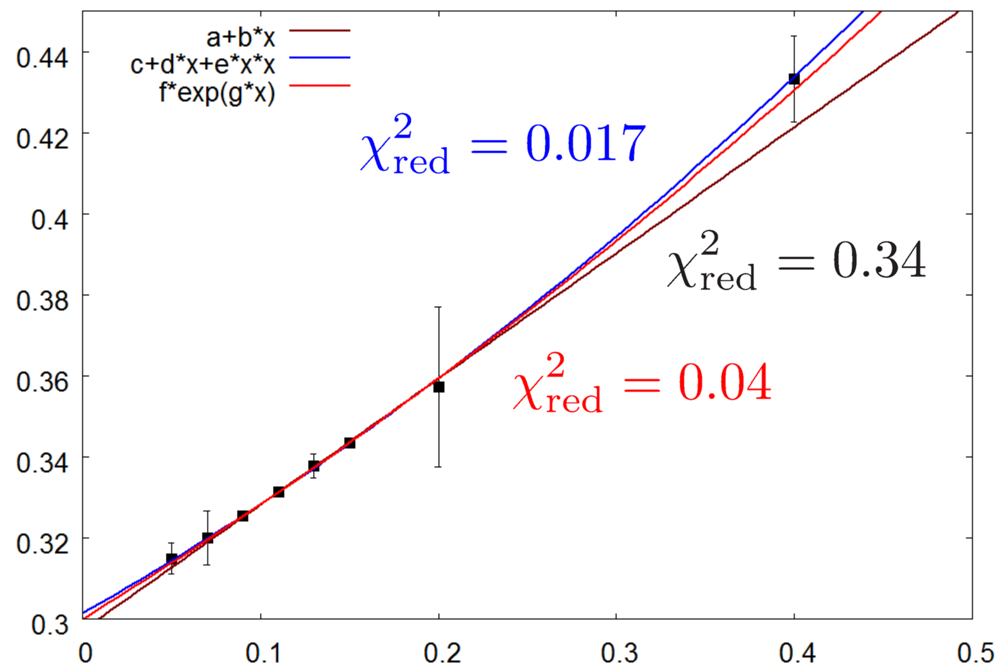
beschreibt die Daten recht gut!

aber auch

g(x) = c + d*x + e*x*x

beschreibt die Daten recht gut!

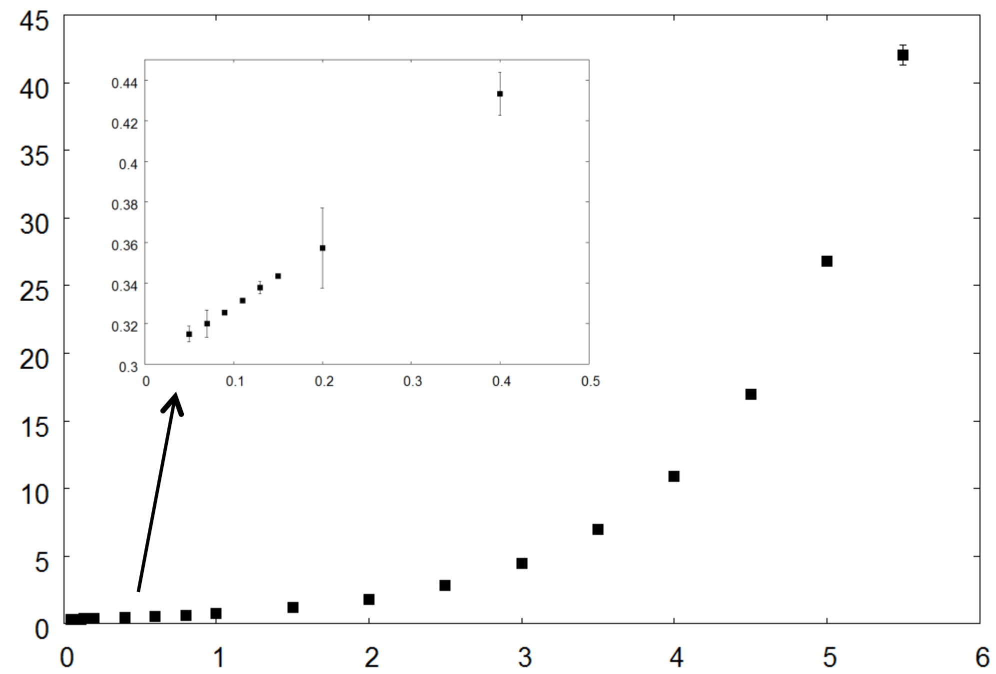
h(x) = f*exp(g*x)   ebenso!



Problem: Innerhalb des kleinen Wertebereichs werden die Daten durch alle Ansätze gut beschrieben!

→ Vergrößerung des Wertebereichs

Weiteres Beispiel



Problem: Innerhalb des kleinen Wertebereichs werden die Daten durch alle Ansätze gut beschrieben!

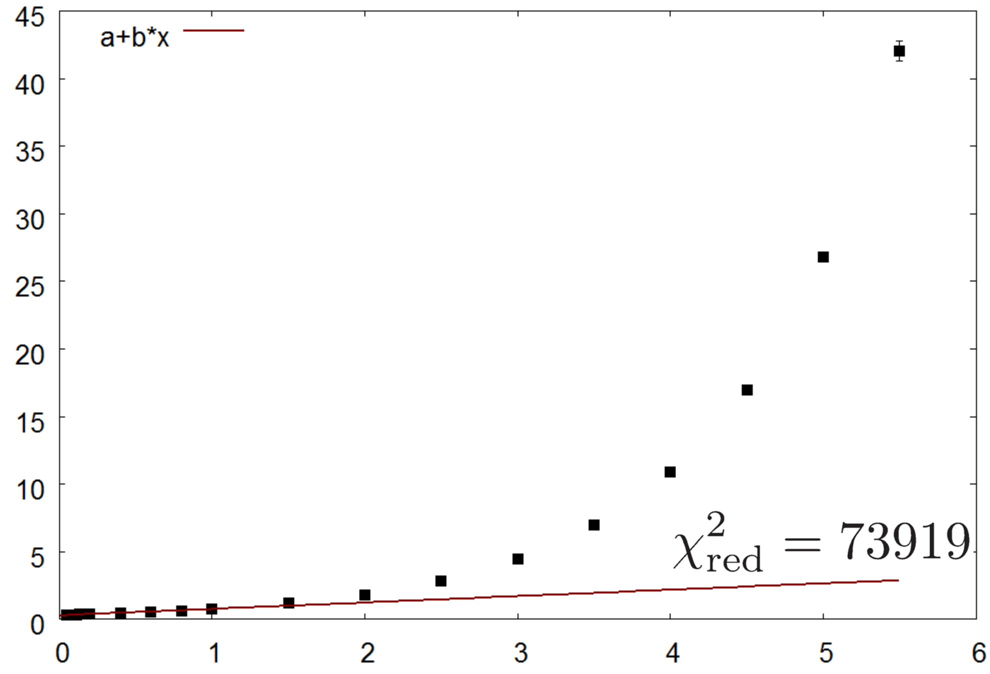
→ Vergrößerung des Wertebereichs

Weiteres Beispiel

Zeigen diese Daten auch einen linearen Zusammenhang?

f(x) = a + b*x

beschreibt die Daten nur für kleine x!



Weiteres Beispiel

Zeigen diese Daten auch einen linearen Zusammenhang?

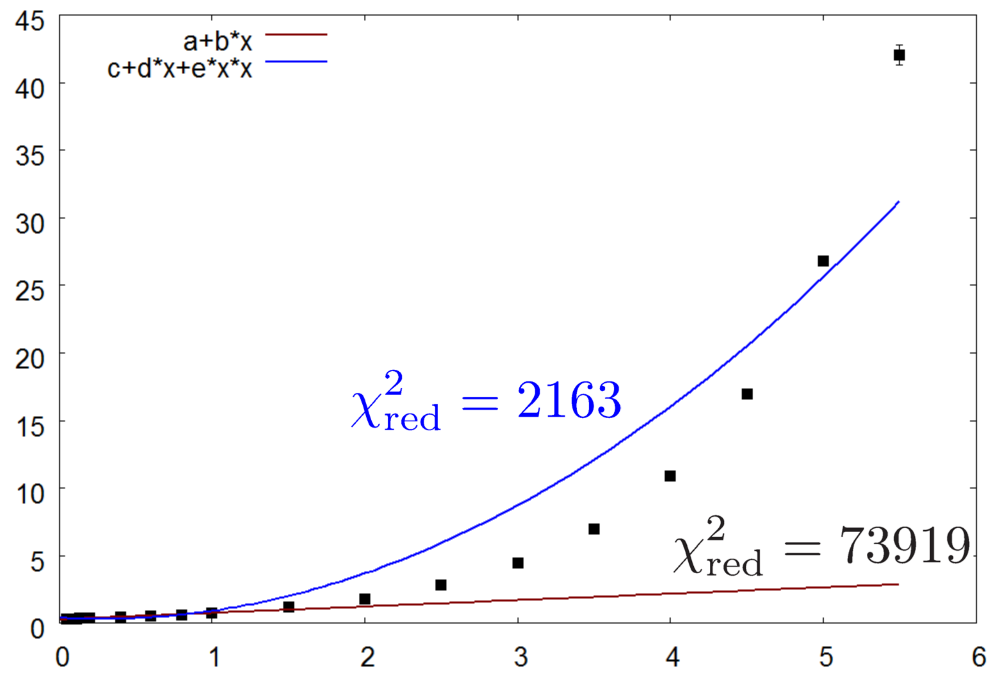
f(x) = a + b*x

beschreibt die Daten nur für kleine x!

aber auch

g(x) = c + d*x + e*x*x

beschreibt die Daten nur für kleine x!



Weiteres Beispiel

Zeigen diese Daten auch einen linearen Zusammenhang?

f(x) = a + b*x

beschreibt die Daten nur für kleine x!

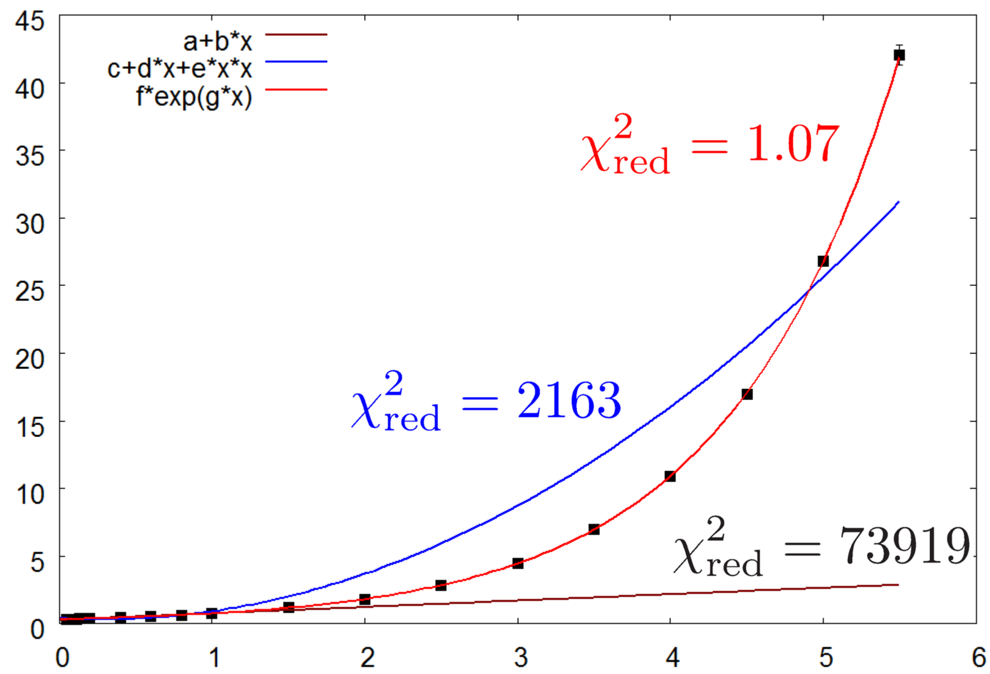
aber auch

g(x) = c + d*x + e*x*x

beschreibt die Daten nur für kleine x!

h(x) = f*exp(g*x)

beschreibt die Daten im gesamten Wertebereich!



nicht-lineare Abhängigkeit

Bsp.:

Messung der

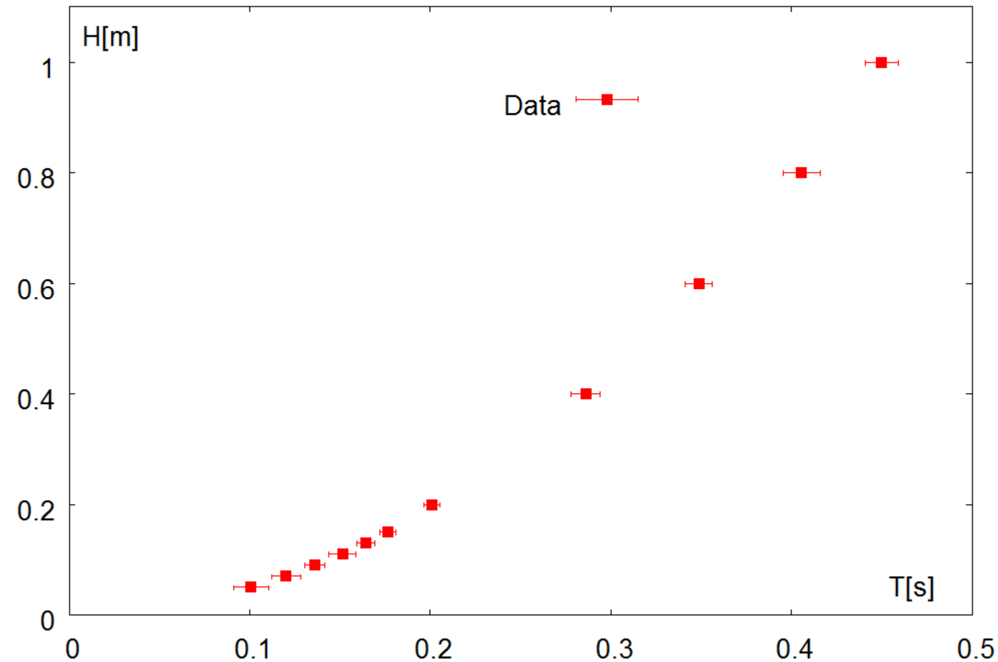
Gravitationsbeschleunigung g

durch Bestimmung der Fallzeit T

einer Kugel in Abhängigkeit von der

Fallhöhe H:

$$H(T) = \frac{1}{2} g T^2$$



nicht-lineare Abhängigkeit

Bsp.:

Messung der

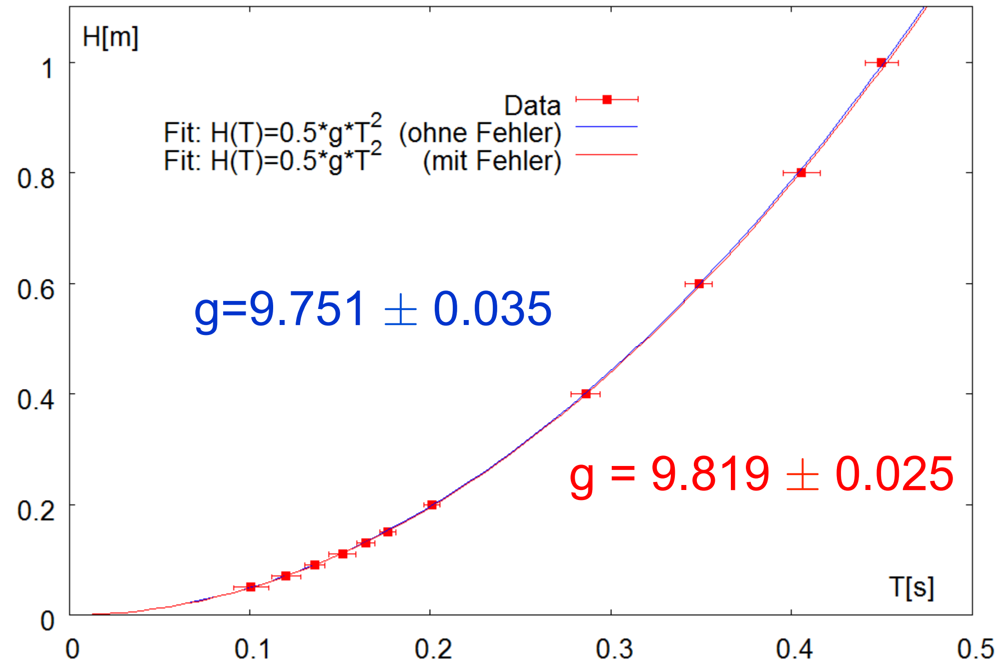
Gravitationsbeschleunigung g

durch Bestimmung der Fallzeit T

einer Kugel in Abhängigkeit von der

Fallhöhe H:

$$H(T) = \frac{1}{2} g T^2$$



Linearisierung der Daten

Bsp.:

Messung der

Gravitationsbeschleunigung g

durch Bestimmung der Fallzeit T

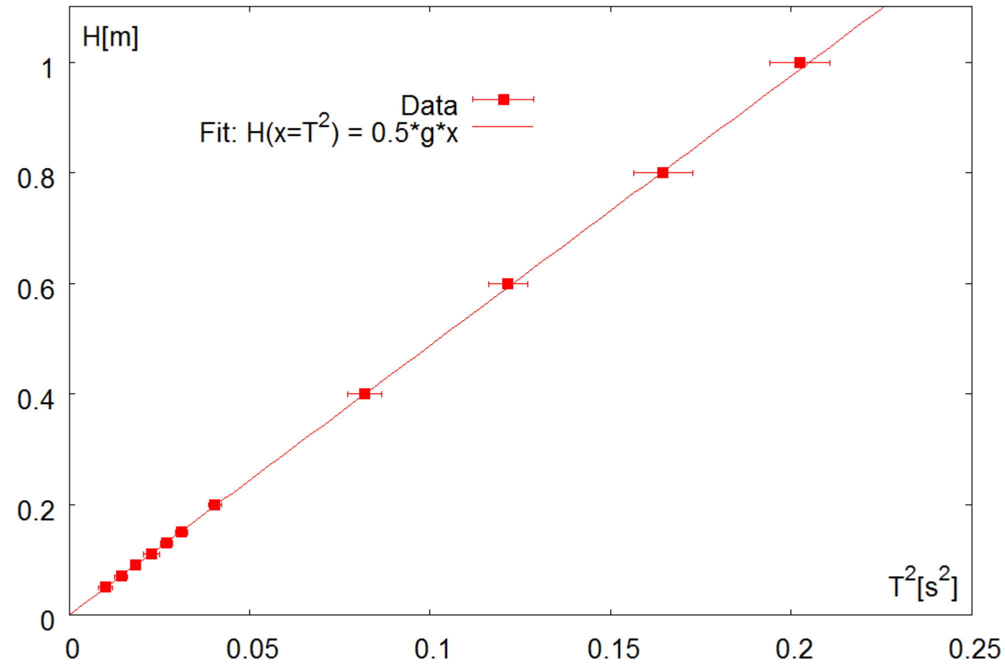
einer Kugel in Abhängigkeit von der

Fallhöhe H:

$$H(T) = \frac{1}{2} g T^2$$

Benutze $x=T^2$ auf der x-Achse:

$$H(T) = \frac{1}{2} g x$$



Systematische Fehler

Hier scheint die lineare Abhängigkeit

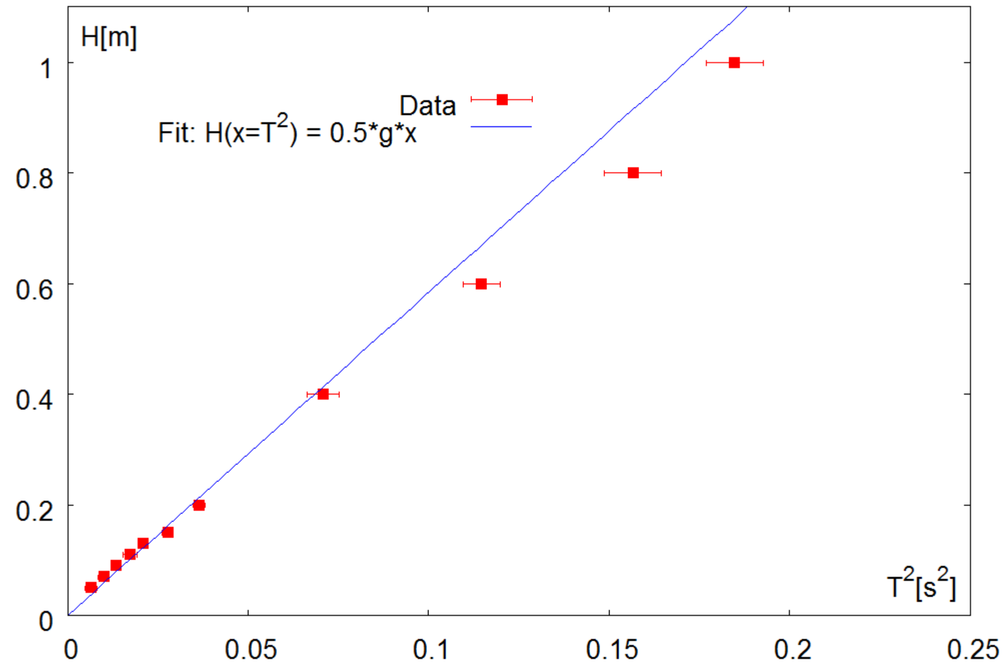
nicht zu passen!?

→ evtl. auf Grund von systematischen Fehlern oder Messfehlern

Daher sollte man im Ansatz eine

Konstante benutzen:

$$H(T) = \frac{1}{2} g T^2 + const$$



Systematische Fehler

Hier scheint die lineare Abhängigkeit

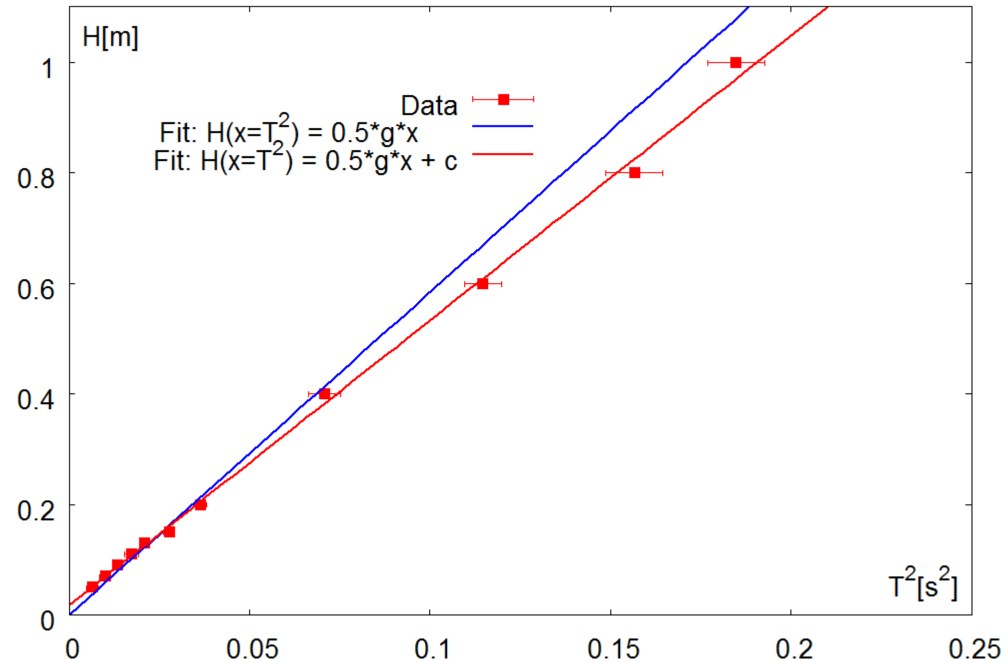
nicht zu passen!?

→ evtl. auf Grund von systematischen
Fehlern oder Messfehlern

Daher sollte man im Ansatz eine

Konstante benutzen:

$$H(T) = \frac{1}{2} g T^2 + const$$



Ist die Konstante innerhalb der Fehler mit Null verträglich, so entsprechen die Daten Der theoretischen Vorhersage.

Ansonsten sollte man untersuchen und diskutieren woher die Abweichung kommt!

Transformationen

Bsp.:

Messung der

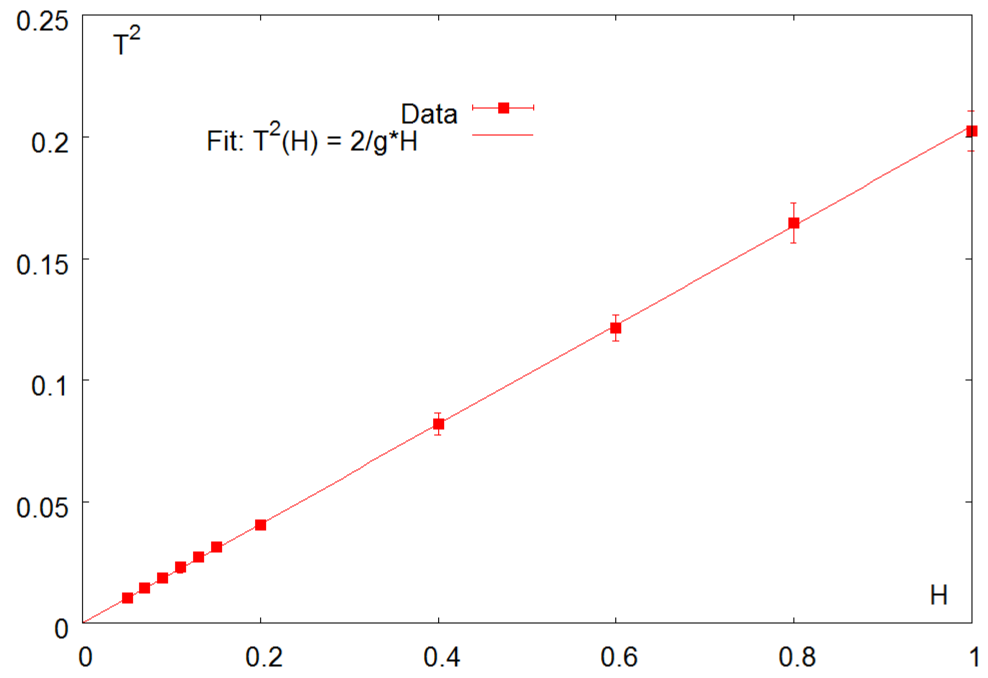
Gravitationsbeschleunigung g

durch Bestimmung der Fallzeit T

einer Kugel in Abhängigkeit von der

Fallhöhe H:

$$H(T) = \frac{1}{2} g T^2$$



Umstellen der Gleichung H <-> T^2:

$$T^2(H) = 2/g * H$$

Transformationen

Bsp.:

Messung der

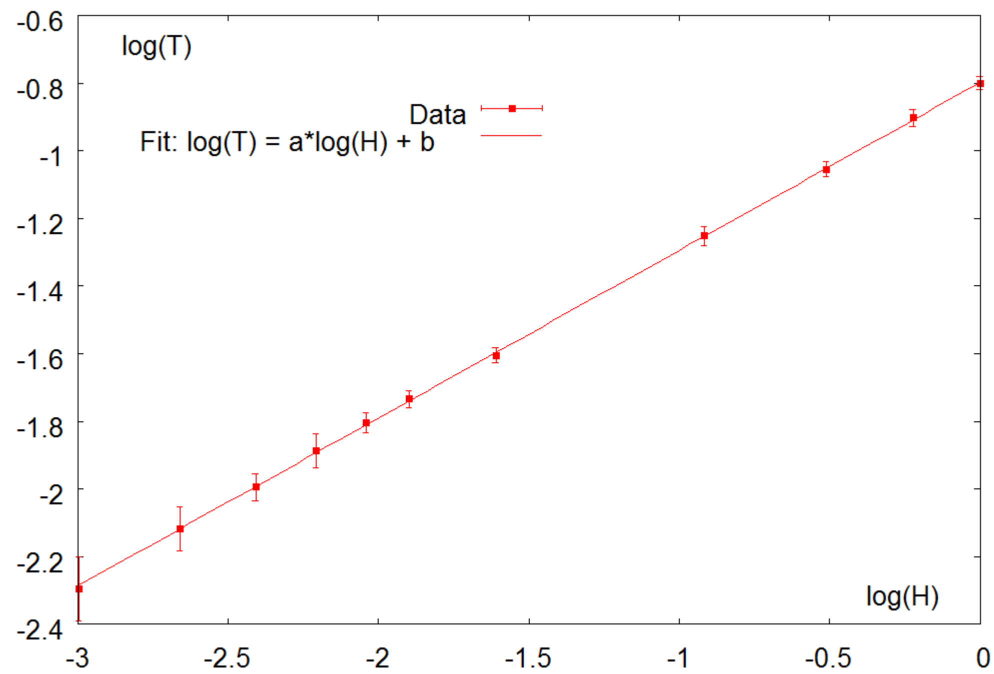
Gravitationsbeschleunigung g

durch Bestimmung der Fallzeit T

einer Kugel in Abhängigkeit von der

Fallhöhe H:

$$H(T) = \frac{1}{2} g T^2$$



Umstellen der Gleichung H <-> T^2:

$$\begin{aligned} T^2(H) &= 2/g * H \\ \log(T) &= 1/2 * \log(H) + 1/2 * \log(2/g) \end{aligned}$$

Einschub: Fehlerfortpflanzung

Messpunkte $(x_1, y_1 \pm \sigma_1), \ldots (x_N, y_N \pm \sigma_N)$

Die Daten sollen mit einer Funktion f(y_i) transformiert werden!

Wie ändern sich die Fehler dabei?

Allgemein: $f(x_1, x_2, \ldots)$

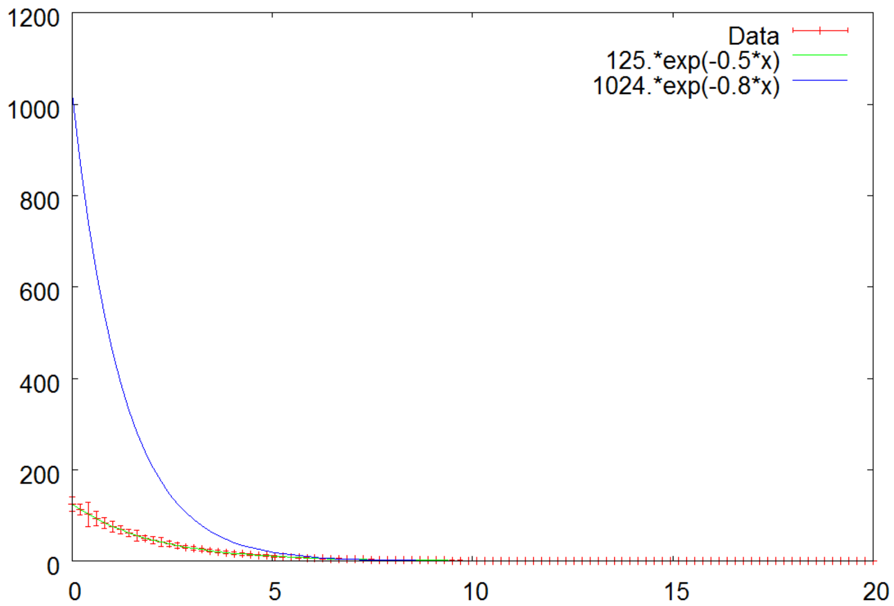
Maximalfehler: $\Delta f = |\frac{\partial f}{\partial x_1}|\Delta x_1 + |\frac{\partial f}{\partial x_2}|\Delta x_2 + \cdots$

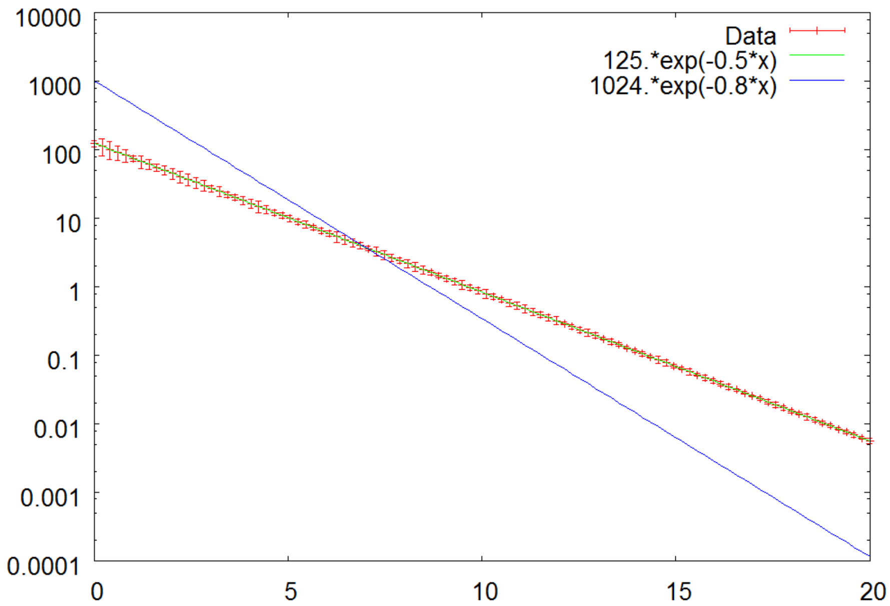
Gaußsche Fehlerfortpflanzung:

$$\Delta f = \sqrt{\left(\frac{\partial f}{\partial x_1}\Delta x_1\right)^2 + \left(\frac{\partial f}{\partial x_2}|\Delta x_2\right)^2 + \cdots}$$

(genauere Besprechung in einer der nächsten Vorlesungen)

Logarithmische Skalen





Durch den starken exponentiellen Abfall sind die Daten bei großen x-Werten kaum sichtbar.

Ob es wirklich ein exponentieller Zusammenhang ist, ist schwer zu erkennen

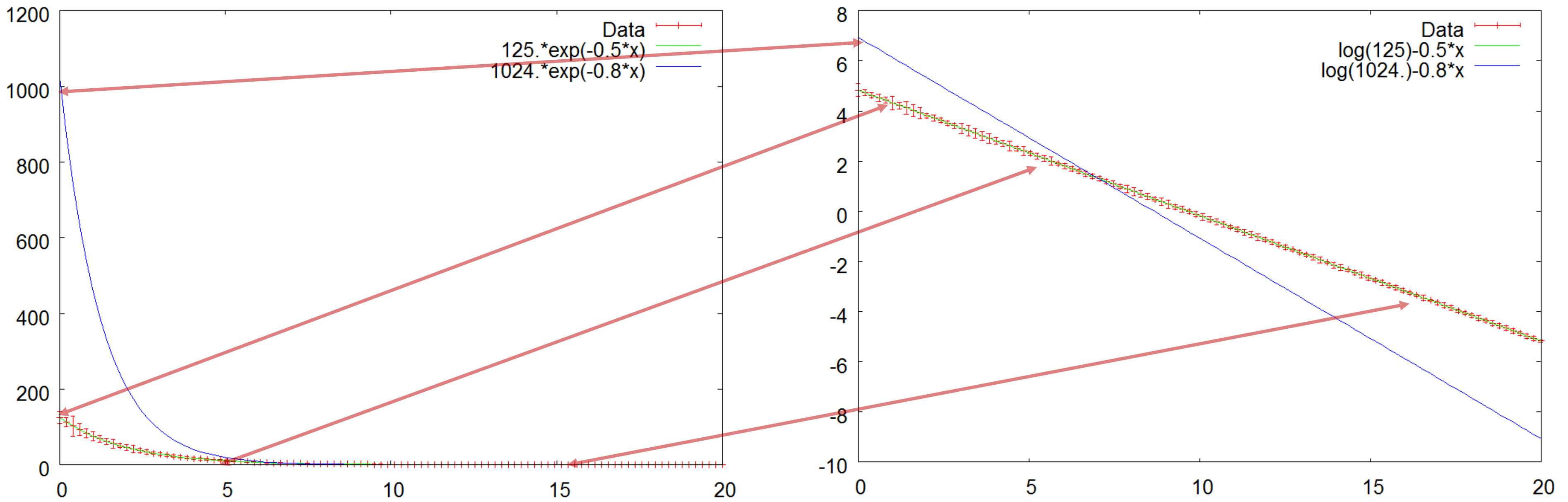
Zur guten Darstellung aller Werte über verschiedene Größenordnungen hinweg ist eine logarithmischen Skala sinnvoller!

Der exponentielle Verlauf wird zu einem linearen Verlauf.

exp. Verlauf exp(m*x) → Steigung m*x

Logarithmische Skalen

zunächst halb-logarithmisch: x-Achse linear und y-Achse logarithmisch



Daten (x_i , $y_i \pm \Delta y_i$), ...

Log-Daten (x_i , $\log(y_i) \pm \Delta y_i / y_i$), ...

Fehlerfortpflanzung für log(y):

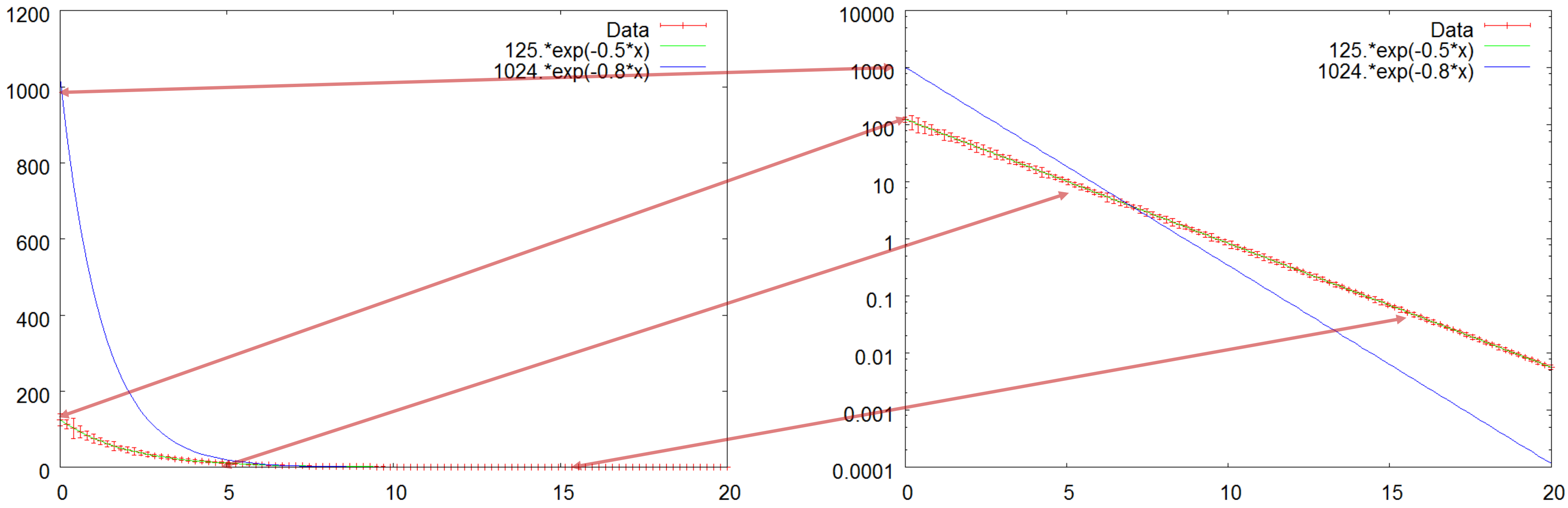
$$\begin{aligned} \Delta log(y) &= \left|\frac{\partial log(y)}{\partial y}\right| \Delta y \\ &= \frac{\Delta y}{|y|} \end{aligned}$$

Vorteil: hier wieder Linearisierung

f(x) = a*exp(b*x)   wird zu

log f(x) = log(a) + b * x

Logarithmische Skalen



Daten (x_i , $y_i \pm \Delta y_i$), ...

set logscale y

plot “datenfile“ using 1:2:3 with errors

oder: Änderung der Skala

gnuplot: set log y

Zum Plotten ist dies besser!

Zum Fitten besser Linearisierung ausnutzen!

Weiteres Beispiel

Zeigen diese Daten auch einen linearen Zusammenhang?

f(x) = a + b*x

beschreibt die Daten nur für kleine x!

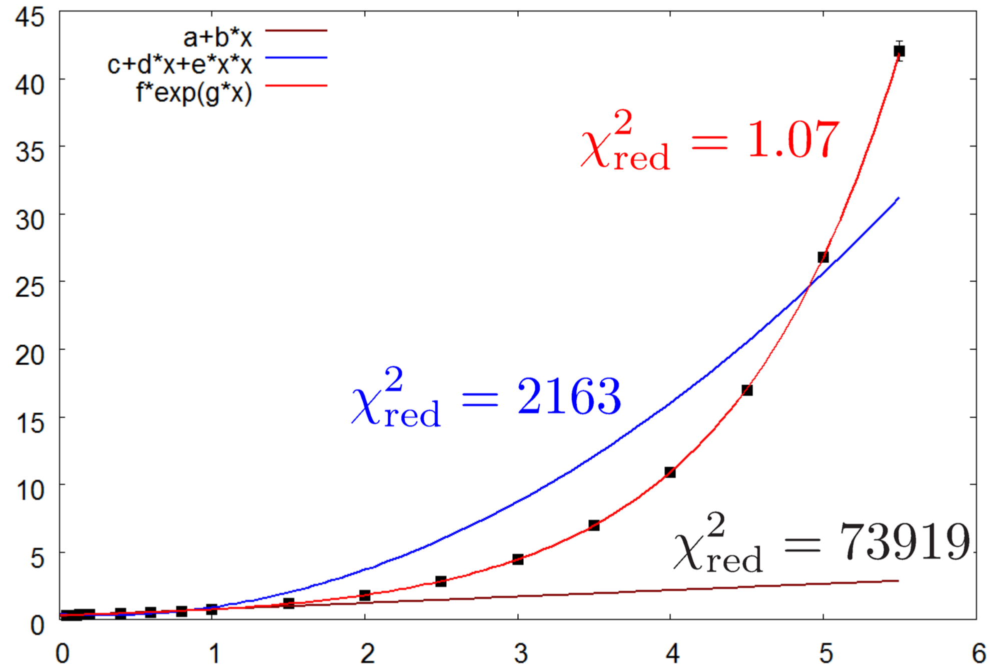
aber auch

g(x) = c + d*x + e*x*x

beschreibt die Daten nur für kleine x!

h(x) = f*exp(g*x)

beschreibt die Daten im gesamten Wertebereich!



Weiteres Beispiel

Zeigen diese Daten auch einen linearen Zusammenhang?

f(x) = a + b*x

beschreibt die Daten nur für kleine x!

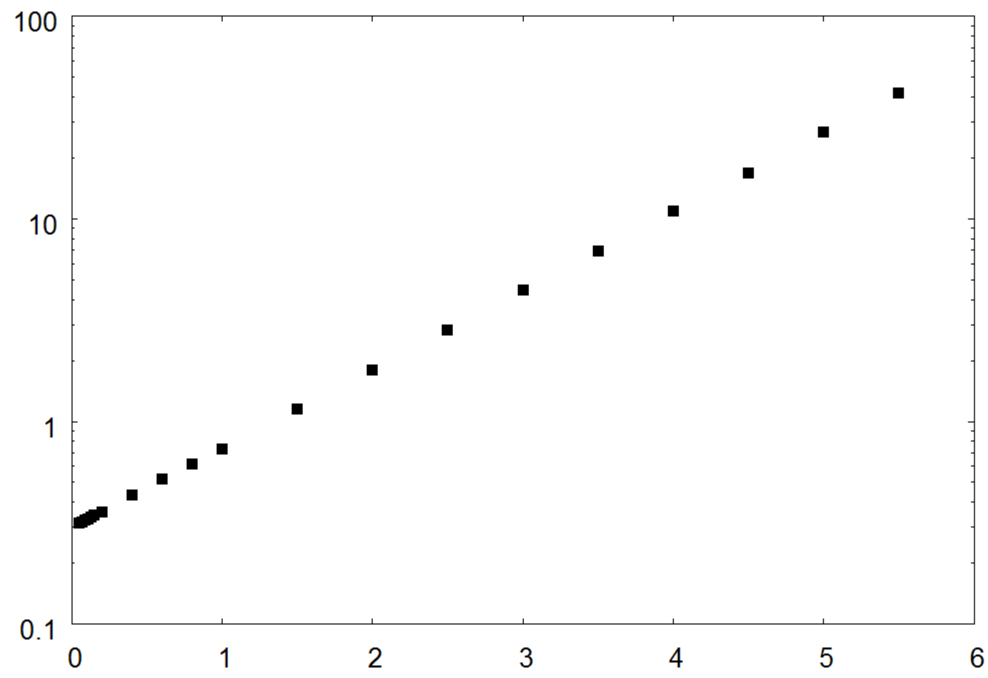
aber auch

g(x) = c + d*x + e*x*x

beschreibt die Daten nur für kleine x!

h(x) = f*exp(g*x)

beschreibt die Daten im gesamten Wertebereich!

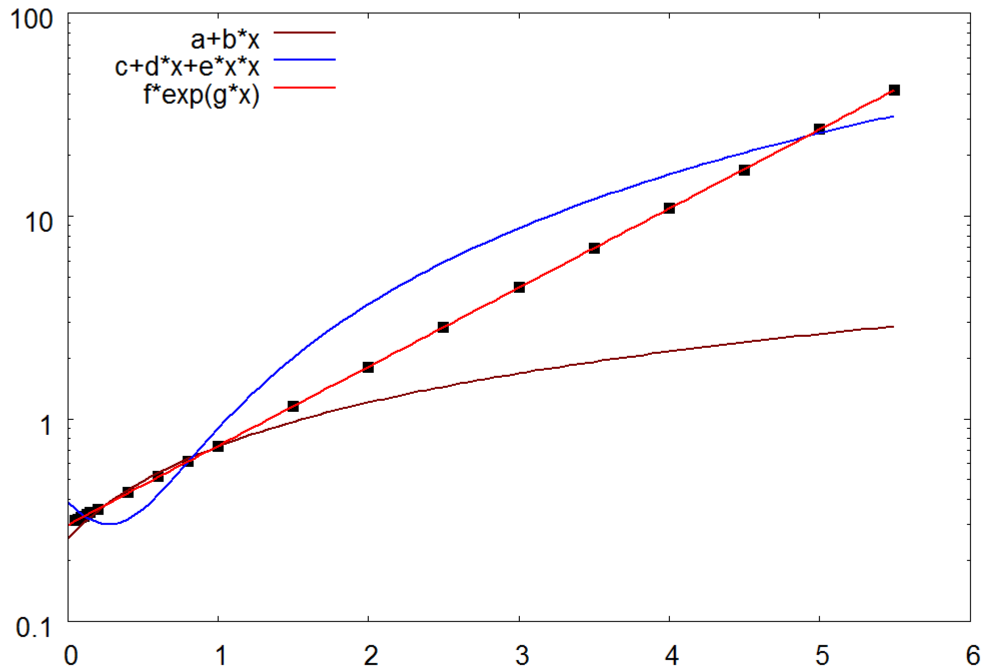


Weiteres Beispiel

Zeigen diese Daten auch einen linearen Zusammenhang?

f(x) = a + b*x

beschreibt die Daten nur für kleine x!



aber auch

g(x) = c + d*x + e*x*x

beschreibt die Daten nur für kleine x!

h(x) = f*exp(g*x) → linear auf logarithmischer Skala

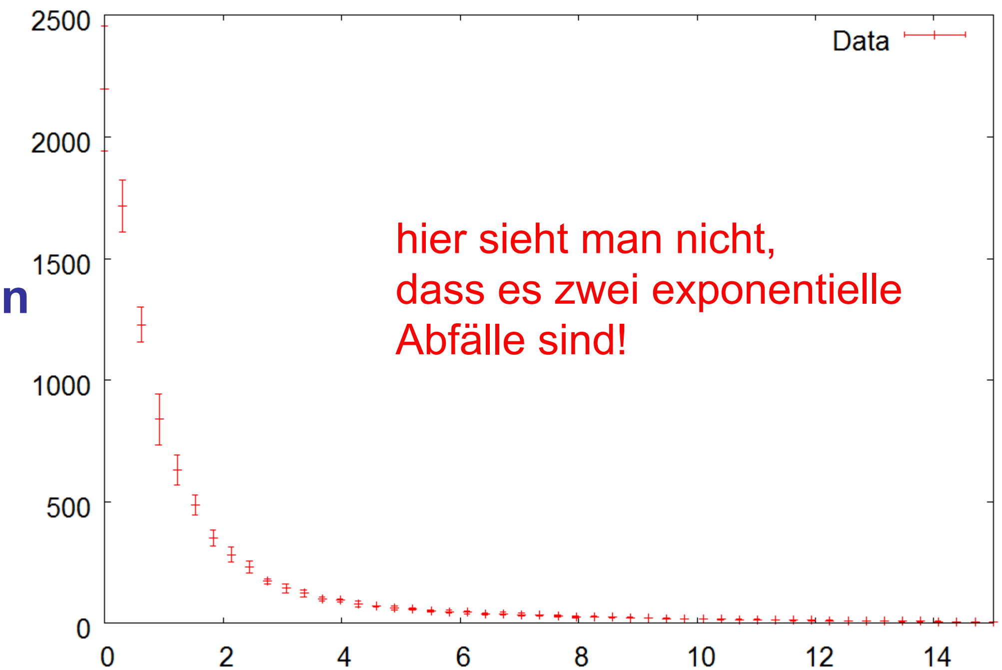
beschreibt die Daten im gesamten Wertebereich!

Logarithmische Skalen

ein komplizierteres Beispiel:

Summe zweier Exponentialfunktionen

$$N(t) = N_1 e^{-\lambda_1 t} + N_2 e^{-\lambda_2 t}$$



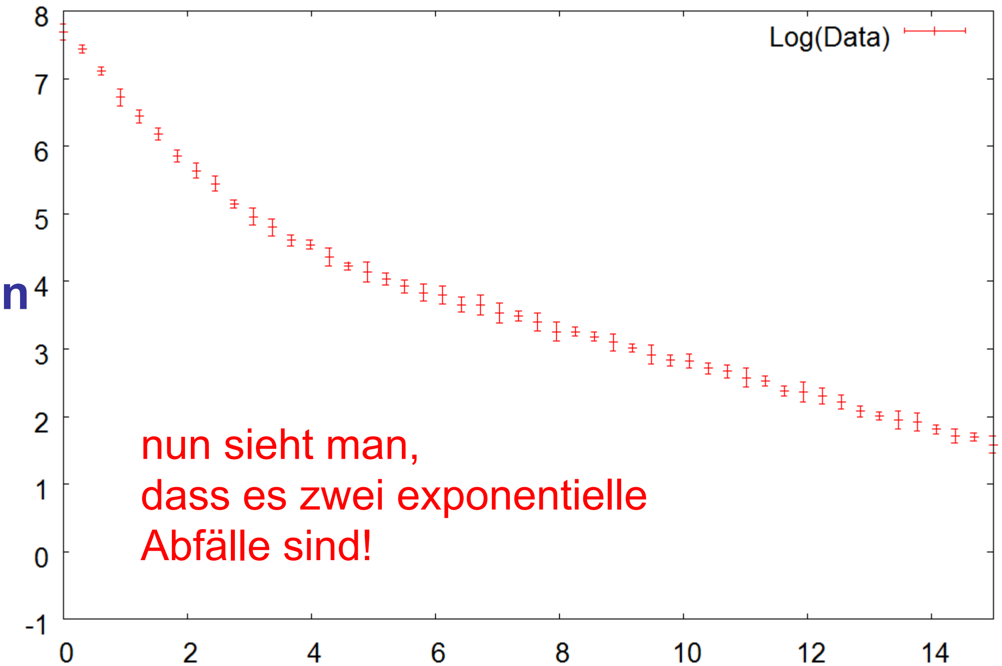
Beispielsweise beim radioaktiven Zerfall zweier Isotope.
N(t) ist die Zählrate von Elektronen in einem Zählrohr.
λ_1 und λ_2 die Zerfallskonstanten der beiden Isotope, λ_i = (ln 2)/T_i,
wobei T_i die Halbwertszeiten sind.

Logarithmische Skalen

ein komplizierteres Beispiel:

Summe zweier Exponentialfunktionen

$$N(t) = N_1 e^{-\lambda_1 t} + N_2 e^{-\lambda_2 t}$$



Annahme: $\lambda_2 < \lambda_1$

für große t: $\exp(-\lambda_2 t) \gg \exp(-\lambda_1 t)$ und die zweite Exponentialfunktion dominiert

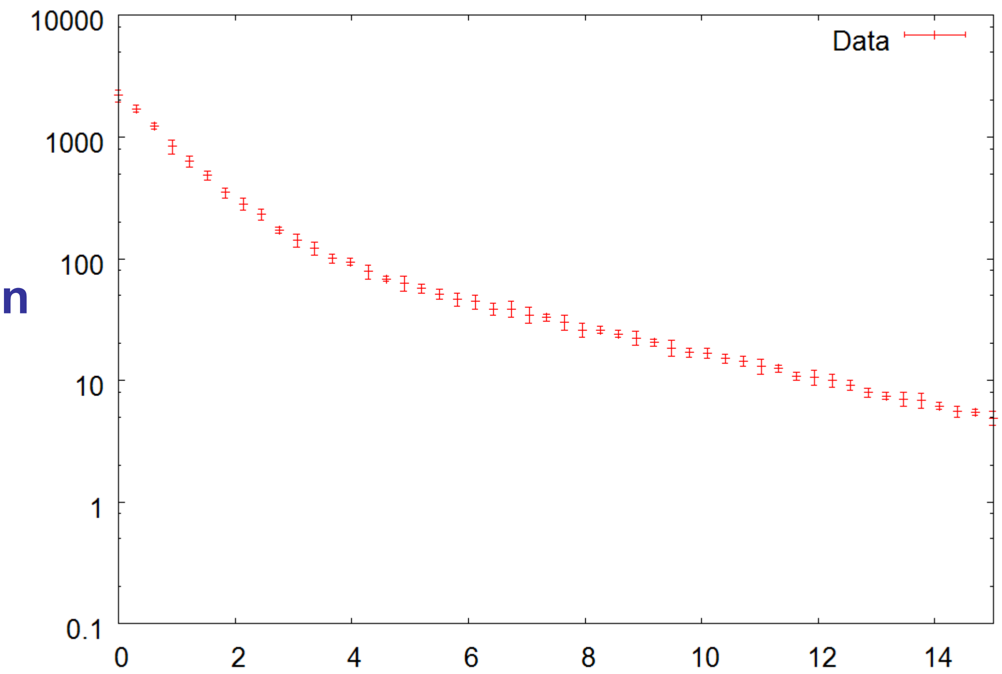
für kleine t: $\exp(-\lambda_1 t) \gg \exp(-\lambda_2 t)$ und die erste Exponentialfunktion dominiert

Logarithmische Skalen

ein komplizierteres Beispiel:

Summe zweier Exponentialfunktionen

$$N(t) = N_1 e^{-\lambda_1 t} + N_2 e^{-\lambda_2 t}$$



Fit mit 4 Parametern?

- instabil und abhängig von Startwerten der Parameter

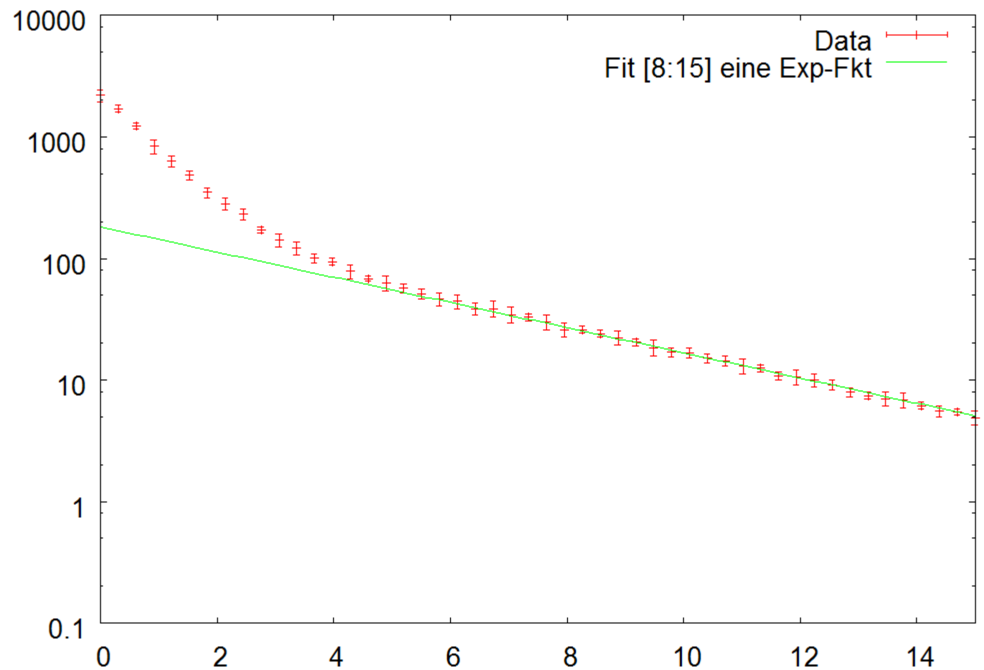
- Abschätzung der Startwerte schwierig

Logarithmische Skalen

ein komplizierteres Beispiel:

Summe zweier Exponentialfunktione

$$N(t) = N_1 e^{-\lambda_1 t} + N_2 e^{-\lambda_2 t}$$

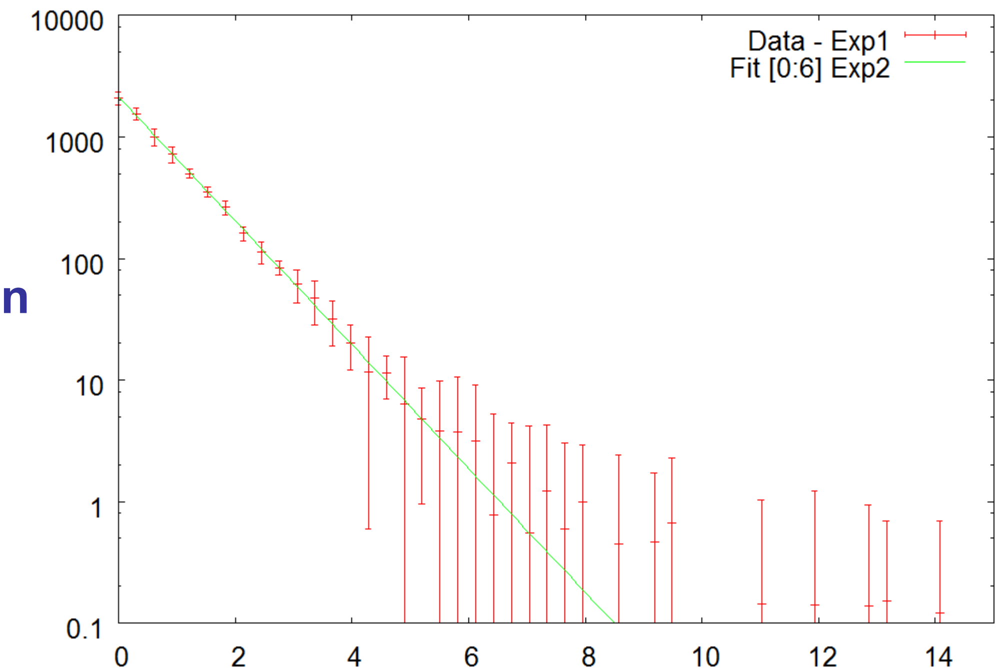


Fit zunächst mit einer Exponentialfunktion bei großen t:

gnuplot: n2(t) = log(N2) – l2*t

fit [8:15] n2(x) “Data“ u 1:(log($2)):($3/$2) via N2,l2

Logarithmische Skalen



ein komplizierteres Beispiel:

Summe zweier Exponentialfunktionen

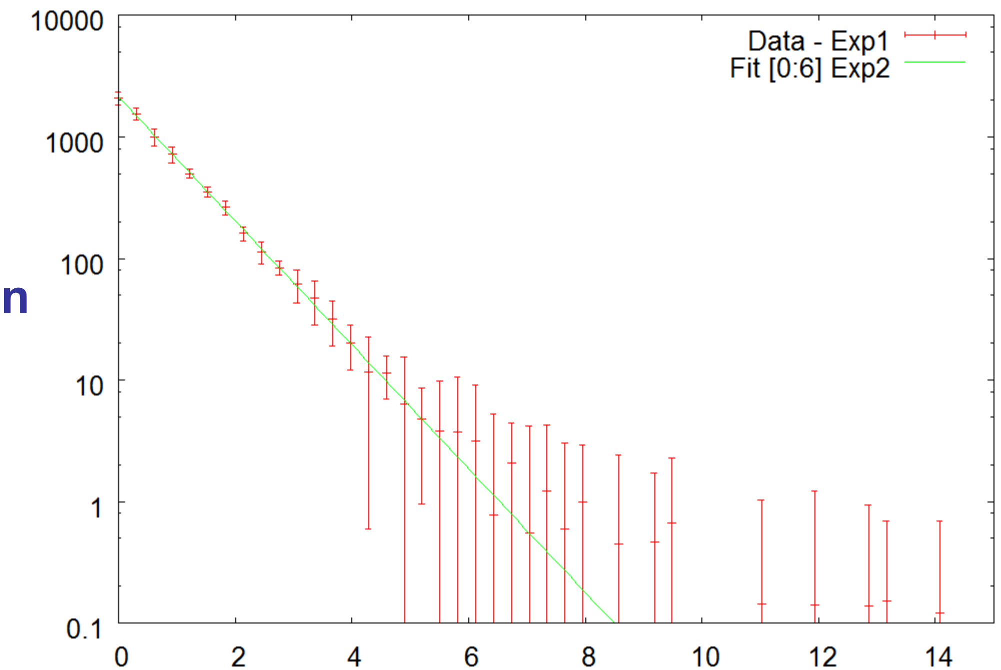
$$N(t) = N_1 e^{-\lambda_1 t} + N_2 e^{-\lambda_2 t}$$

Subtrahiere das Ergebnis für n2(t) von den Daten:

gnuplot: n2(t) = log(N2) – l2*t
fit [8:15] n2(x) “Data“ u 1:(log($2)):($3/$2) via N2,l2
plot “Data“ u 1:($2-n2(x)):($3) w e

Logarithmische Skalen

ein komplizierteres Beispiel:



Summe zweier Exponentialfunktionen

$$N(t) = N_1 e^{-\lambda_1 t} + N_2 e^{-\lambda_2 t}$$

Fit mit einer Exponentialfunktion bei kleinen t:

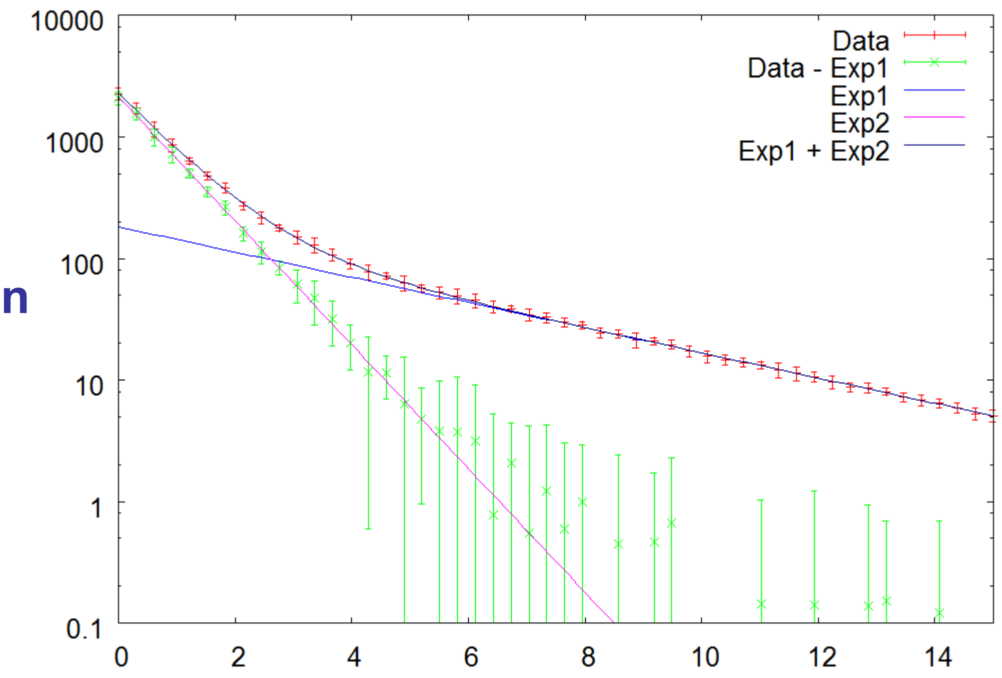
gnuplot: n2(t) = log(N2) – l2*t
fit [8:15] n2(x) “Data“ u 1:(log($2)):($3/$2) via N2,l2
plot “Data“ u 1:($2-n2(x)):($3) w e
n1(t) = log(N1) – l1*t
fit [0:6] n1(x) “Data“ u 1:(log($2-n2(x))):($3/$2) via N1,l1

Logarithmische Skalen

ein komplizierteres Beispiel:

Summe zweier Exponentialfunktionen

$$N(t) = N_1 e^{-\lambda_1 t} + N_2 e^{-\lambda_2 t}$$



Vorteile: Es sind nur lineare Fits nötig
jeweils nur zwei Parameter für die Fits
und Fehlerberechnung ist einfacher
Fits mit vielen Parametern meist instabil
und Fehlerabschätzung schwieriger

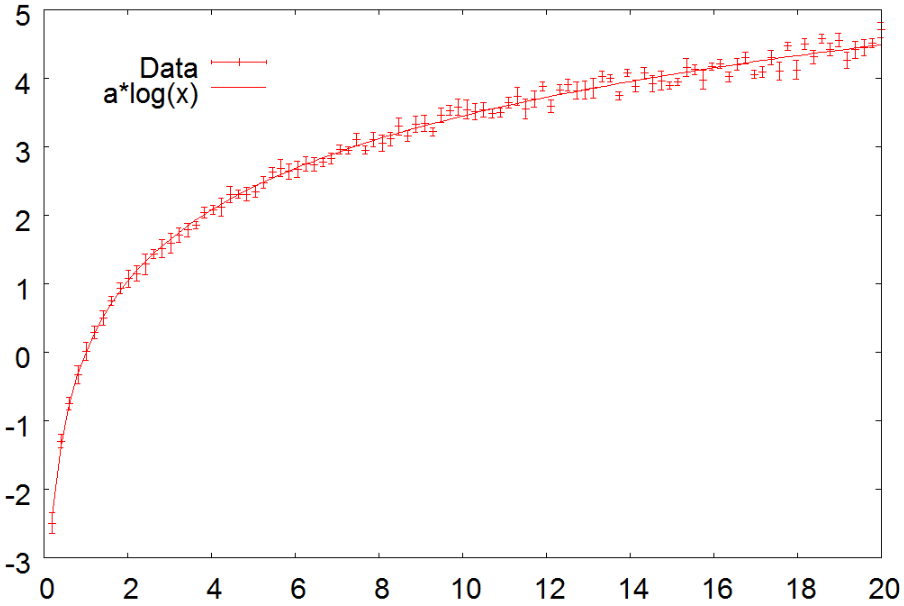
Logarithmische Skalen

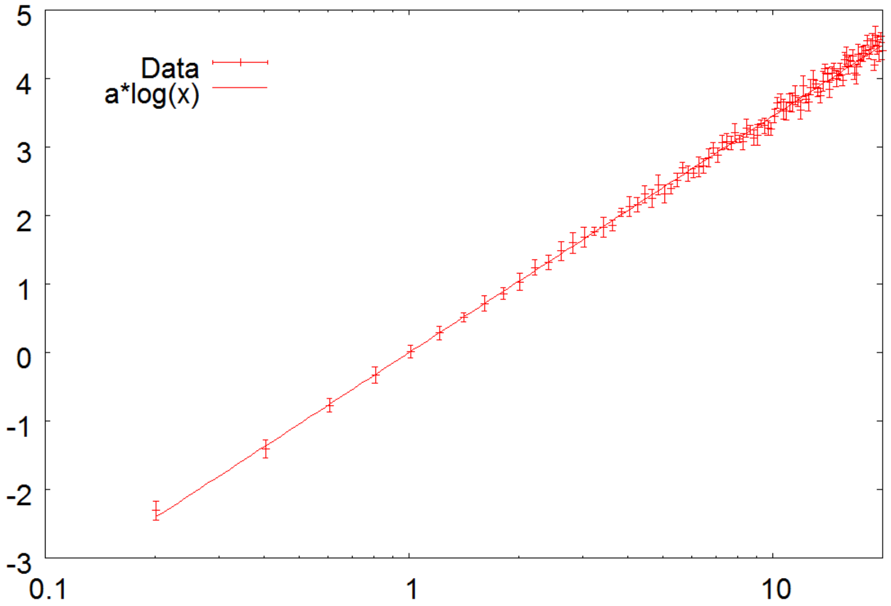
halb-logarithmisch: x-Achse logarithmisch und y-Achse linear

$$y(x) = a * \log(x)$$

$$x' = \log(x)$$

$$y(x) = a * x'$$





Logarithmische Skalen

doppel-logarithmisch: x-Achse logarithmisch und y-Achse logarithmisch

$$y(x) = a * x^p$$

$$\log y(x) = \log(a) + p\, x' \qquad (x' = \log(x))$$